L8: Entry 1 of 1 File: PGPB Oct 18, 2001

DOCUMENT-IDENTIFIER: US 20010032162 A1
TITLE: Methods and systems for market clearance

Pre-Grant Publication (PGPub) Document Number:
20010032162

Detail Description Paragraph:
[0199] In the fragment list 2450, offer or fragment ID's 2452 may be used to point to a specific offer or to a specific fragment within an offer. An alternative embodiment uses a direct pointer to a fragment. Others may use a direct pointer to an offer and then search for the desired fragment within that offer. Lock status 2454 has at least four states. The first is unlocked. The second is locked, meaning if subsequently unlocked, then attempt to relock on another offer. The third is do-not-relock, meaning locked but do not attempt a relock if unlocked. The fourth is potential-lock, meaning unlocked but could lock if a minimum quantity is achieved.

L9: Entry 1 of 1 File: PGPB Aug 16, 2001

DOCUMENT-IDENTIFIER: US 20010014868 A1
TITLE: SYSTEM FOR THE AUTOMATIC DETERMINATION OF CUSTOMIZED PRICES AND PROMOTIONS

Pre-Grant Publication (PGPub) Document Number:
20010014868

Detail Description Paragraph:
[0019] Relevant definitions of terms for the purpose of this description include: (a.) the contractual terms of an offer that one party might make to another (such as the first party's obligation to provide a particular product or service, the second party's obligation to pay a particular price in return via a specified or unspecified payment system, and any other present or future obligations imposed upon either party as conditions of the offer, possibly including but not limited to eligibility restrictions, discounts, future rebates, warrantees, frequent flier miles, sweepstakes eligibility, and guarantees of confidentiality), together with the details of the presentation of that offer to the second party, including any surrounding or accompanying product information or advertising material conveyed by such means as text, sound, or graphical images, are collectively termed an "offer", (b.) the party choosing whether to make an offer is termed a "vendor", (c.) the party to which an offer is made, and who may choose to accept or reject the offer, is termed a "shopper", (d.) a digital representation of an offer's attributes, which may also include attributes of the vendor, is termed an "offer profile", (e.) a digital representation of a shopper's attributes is termed a "shopper profile", (f.) a summary of the degree to which a particular shopper likes or dislikes various offer profiles, which summary constitutes part of that shopper's profile, is termed the "offer demand summary" of that shopper, (g.) a profile consisting of a collection of attributes, such that a particular shopper likes offers whose offer profiles are similar to this collection of attributes, is termed a "search profile", (h.) a specific embodiment of the offer demand summary of a shopper as a set of search profiles is termed the "search profile set" of the shopper, (I.) a collection of offers with similar offer profiles is termed a "cluster", j.) an aggregate profile formed by averaging the offer profiles of all offers in a cluster is termed a "cluster profile", (k.) a real number determined by calculating the statistical variance of the offer profiles of all offers in a cluster, is termed a "cluster variance," (l.) a real number determined by calculating the maximum distance between the offer profiles of any two offers in a given cluster, is termed a "cluster diameter".

Detail Description Paragraph:
[0170] Not all point estimates of the topical interest function f(*, *) should be given equal weight as inputs to the smoothing algorithm. Since passive feedback is less reliable than active feedback, point estimates made from passive feedback should be weighted less heavily than point estimates made from active feedback, or even not used at all. In some domains, a shopper's interests may change over time and, therefore, estimates of topical interest that derive from more recent feedback should also be weighted more heavily. A shopper's interests may vary according to mood, so estimates of topical interest that derive from the current session should be weighted more heavily for the duration of the current session, and past

estimates of topical interest made at approximately the current time of day or on the current weekday should be weighted more heavily. Finally, in domains where shoppers are trying to locate offers of long term interest (automobiles, investments, romantic partners, pen pals, employers, employees, suppliers, service contracts) from the possibly meager information provided by the offer profiles, the shoppers are usually not in a position to provide reliable immediate feedback on an offer, but can provide reliable feedback at a later date. An estimate of topical interest f(V, Y) should be weighted more heavily if shopper V has had more experience with offer Y. Indeed, a useful strategy is for the system to track long term feedback for such offers. For example, if offer profile Y was created in 1990 to describe a particular investment that was available in 1990, and that was purchased in 1990 by shopper V, then the system solicits relevance feedback from shopper V in the years 1990, 1991, 1992, 1993, 1994, 1995, etc., and treats these as successively stronger indications of shopper V's true interest in offer profile Y, and thus as indications of shopper V's likely interest in new investments whose current profiles resemble the original 1990 offer profile Y. In particular, if in 1994 and 1995 shopper V is well disposed toward his or her 1990 purchase of the investment described by offer profile Y, then in those years and later, the system tends to recommend additional investments when they have profiles like offer profile Y, on the grounds that they too will turn out to be satisfactory in 4 to 5 years. It makes these recommendations both to shopper V and to shoppers whose investment portfolios and other attributes are similar to shopper V's. The relevance feedback provided by shopper V in this case may be either active (feedback=satisfaction ratings provided by shopper V) or passive (feedback=difference between average annual return of the investment and average annual return of the Dow Jones index portfolio since purchase of the investment, for example).

L10: Entry 1 of 1 File: PGPB Aug 16, 2001

DOCUMENT-IDENTIFIER: US 20010014868 A1
TITLE: SYSTEM FOR THE AUTOMATIC DETERMINATION OF CUSTOMIZED PRICES AND PROMOTIONS

Pre-Grant Publication (PGPub) Document Number:
20010014868

Detail Description Paragraph:
[0020] This system teaches a variety of related techniques relevant to collecting and using profiles of shoppers, promotions, and products to increase the efficiency and profitability of on-line shopping. The following sections describe the implementation of the basic on-line price point system in detail, including customized price points and promotions, custom coupons, and custom construction of products such as insurance or investment portfolios. The architecture of the shopping system is covered first, then detail is given on how profiles of offers and shoppers are created, compared and clustered. The final set of sections then describe applications of the method: automatically selecting offers to maximize vendor profit, use of custom coupons, joint promotions of multiple items and construction of custom offers, shopper's agents and buyers clubs, and the use of profiles for enhancing off-line sales.

Detail Description Paragraph:
[0292] The same profiling and clustering techniques described above can also be used more generally to match shoppers with vendors or other shoppers who have complementary interests. There are many situations in commerce where it is useful to match up multiple people with similar interests: shoppers can be matched to buy and sell items, to barter and exchange items, to wager with each other about sporting events, to place bids on an item(s) being auctioned to hedge risk, to get lower prices by purchasing in bulk, or to discuss their common interests. A group of shoppers with similar shopper profiles or offer demand summaries can be thought of as a buyers' club or a `mini`-market that is assembled automatically, on an ad hoc basis.

Detail Description Paragraph:
[0293] The buyers' club subsystem attempts to identify groups of shoppers with common interests. These groups, herein termed "pre-clubs," are represented as sets of shoppers. Whenever the buyers' club subsystem identifies a pre-club, it will subsequently attempt to put the users in said pre-club in contact with each other, as described below. Each pre-club is said to be "determined" by a cluster of messages, pseudonymous users, search profiles, or target objects. To identify pre-clubs, shoppers are clustered by the similarity of their profiles, using for example k-means clustering or soft clustering, and every cluster constitutes a pre-club. If each shopper has an associated search profile set, a better method is available: all search profiles of all pseudonymous users can be clustered based on their similarity, and each cluster of search profiles determines a pre-club whose members are the shoppers from whose search profile sets the search profiles in the cluster are drawn. Each such pre-club is a group of shoppers who are interested in offers with a particular type of profile, and so presumably share an interest. Once the buyers' club subsystem identifies a cluster C of shopper profiles or search

profiles that determines a pre-club M, it attempts to arrange for the members of this pre-club to have the chance to participate in a common buyers' club V. In many cases, an existing buyers' club V may suit the needs of the pre-club M. The buyers' club subsystem first attempts to find such an existing club V. In the case where cluster C is a cluster of shopper profiles, V may be chosen to be any existing buyers' club such that the cluster profile of cluster C is within a threshold distance of the mean shopper profile of the active members of buyers' club V; in the case where the cluster C is a cluster of search profiles, V may be chosen to be any existing buyers' club such that the cluster profile of cluster C is within a threshold distance of the cluster profile of the largest cluster resulting from clustering all the search profiles of active members of buyers' club V. The threshold distance used in each case is optionally dependent on the cluster variance or cluster diameter of the profile sets whose means are being compared.

Detail Description Paragraph:
[0315] A method has been described for the customized determination of which products a purchaser would be most likely to buy, and which offering price and promotions (coupons, advertisements) can be expected to maximize the vendors profitability. In particular, the system automatically constructs profiles of the shoppers based on their demographics, and history of information request and purchases. The shoppers' behaviors in response to product advertisements or other promotions are then predicted by finding what the other shoppers with the most similar profiles have done. "Rapid profiling" techniques can be used to characterize the shopper with a minimum number of initial questions; shopper profiles are then automatically updated as their on-line shopping is monitored. Additionally, we present similar profile-based methods for custom construction of products such as insurance or investment portfolios, for custom electronic shopping mall layout, and for automatic construction of buyers' clubs for commerce. These buyers' clubs may either be groups of shoppers and vendors wishing to trade with one another, or groups of shoppers wishing to share expertise. These methods of suggesting products, prices, and promotions can also be used in conjunction with smartcards and with electronic cash. Finally, the profiles developed on-line can be used to devise off-line sales and marketing strategies.

Previous Doc   Next Doc   Go to Doc#

L11: Entry 1 of 2 File: PGPB Oct 18, 2001

DOCUMENT-IDENTIFIER: US 20010032162 A1
TITLE: Methods and systems for market clearance

Pre-Grant Publication (PGPub) Document Number:
20010032162

Summary of Invention Paragraph:
[0014] An RFQ is an offer to buy that is published to many prospective sellers. Sellers bid for the business. The buyer typically chooses the seller based on price and other criteria. RFQ's may or may not constitute a binding offer to buy. RFQ's and RFP's are popular commerce vehicles for large buyers (e.g., governments and large corporations) who have a sufficiently large order to attract sellers and to justify the cost of publishing the RFQ.

Summary of Invention Paragraph:
[0022] The present invention enables the creation of an exchange that may be run by a marketplace operator that may be a participant, a third party, or a technology provider. Exchanges consistent with the present invention comprise a meeting place where buyers and sellers can efficiently find each other, make individual offers to buy and sell products with varying attributes at varying prices with varying fulfillment costs, aggregate their collective demand or supply, and produce many-to-many transactions at multiple prices at the same time.

Detail Description Paragraph:
[0103] A product specification can include the totality of the terms the product must meet. So, for example, a product specification may contain attributes related to the seller or buyer (e.g., location, credit rating) or delivery, payment service rating or any other terms needed to fully define the product being transacted.

Detail Description Paragraph:
[0109] FIG. 6 is a block diagram illustrating a pool specification within an offer consistent with the present invention. A single offer is placed into a single pool, and a pool ID, such as pool ID 620, may identify a single pool. Market-clearing systems consistent with the present invention utilize discrete pools with discrete pool close times, where the close times are determined using rules established by the marketplace operator. A pool can be identified by its close event, such as close event 630. A pool close event corresponds to a defined event whose time of occurrence can be determined with precision, for example: 5:00 p.m. PST Wednesday, when 500 offers have locked in this Pool, or 4:00 p.m. GMT on the second day after the last game of the 2001 World Series. A pool specification may also include offer restrictions such as offer restrictions 640. These restrictions may specify offer constraints based on, for example, product attributes, quantities, and buyer and seller attributes. Offers that do not meet these restrictions may not be eligible to join the pool. When a pool specification includes an offer restriction 640, then the pool can be identified by reference to both its close event 630 and its offer restrictions 640.

Detail Description Paragraph:
[0110] Individual marketplace operators may use offer restrictions to impose constraints on pool membership. Marketplace operators may have business reasons to

organize pools according to buyer, seller, product, or other attributes, in addition to the close time. For example, a marketplace operator may require that all offers in a pool come from a certain category of buyer or seller, or that the offers specify products in the same product category. A category can be as broad or narrow as the marketplace operator wishes. For example, a product category could be office machines or copiers or Xerox.RTM. copiers or Xerox small office copiers or Xerox model 4444 copiers or one specific configuration of a Xerox model 4444 copier. Similarly, a buyer category could be used to group buyers on the basis of their creditworthiness, so that credit card payers are in a different pool from major companies with a strong credit rating.

Detail Description Paragraph:
[0111] The marketplace operator sets the rules for transforming requested close times in offers into adjusted close times that match the available pool close times in the marketplace. A marketplace operator may adjust requested close times to the nearest available pool close time, to the pool close time immediately before or after the requested close time, or to any other available pool close time by any other rules. The marketplace operator who chooses to organize pools by close time and other criteria, such as, for example, buyer, seller, product, or other attributes, need not have all pools with differing criteria close at the same times.

Detail Description Paragraph:
[0114] FIG. 8 is a block diagram illustrating a disadvantaged offer price specification consistent with the present invention. Disadvantaged offer price specifications, like specification 810, have a pricing function 820. The pricing function is used to calculate an offering price that is a function of the attributes of the opposite offer. Thus, a seller's price in a buyer-advantaged pool may be a function of the buyer's desired good or service, delivery location, time of delivery, insurance, warranty options, service contracts purchased at time of sale, product features and options, means of payment, credit worthiness, and any other attribute on which the seller chooses to base its offering price. A buyer's price in a seller-advantaged pool may be a function of the seller's desired good or service, location, service quality rating, delivery terms, payment options, warranty, and any other attribute on which the buyer chooses to base its offering price.

Detail Description Paragraph:
[0145] Market-clearing engines 1306, consistent with the present invention, exist within a marketplace system that determines pool criteria, enables buyers and sellers to define straddles, supplies the events that signal the close of a pool, and provides the market-clearing engine with offers to post to pools and to then compare and lock with other offers in the same pool. A market-clearing engine also manages related transactions, including withdrawals, straddles, and reporting.

Previous Doc Next Doc Go to Doc#

L11: Entry 2 of 2 File: PGPB Aug 16, 2001

DOCUMENT-IDENTIFIER: US 20010014868 A1
TITLE: SYSTEM FOR THE AUTOMATIC DETERMINATION OF CUSTOMIZED PRICES AND PROMOTIONS

Pre-Grant Publication (PGPub) Document Number:
20010014868

Detail Description Paragraph:
[0020] This system teaches a variety of related techniques relevant to collecting and using profiles of shoppers, promotions, and products to increase the efficiency and profitability of on-line shopping. The following sections describe the implementation of the basic on-line price point system in detail, including customized price points and promotions, custom coupons, and custom construction of products such as insurance or investment portfolios. The architecture of the shopping system is covered first, then detail is given on how profiles of offers and shoppers are created, compared and clustered. The final set of sections then describe applications of the method: automatically selecting offers to maximize vendor profit, use of custom coupons, joint promotions of multiple items and construction of custom offers, shopper's agents and buyers clubs, and the use of profiles for enhancing off-line sales.

Detail Description Paragraph:
[0291] Similar profiling techniques can be built (e.g. into Internet browsers) which attempt to maximize consumer surplus, rather than the profitability of the vendor. Such consumer agents would be used to locate bargains--where "bargain" is defined from the standpoint of the individual shopper. I.e., given a profile of the shopper, combined with specific attributes of what the shopper is looking for, the consumer agent would search over one or more vendor sites to find items which are particularly appealing. Consumers may also wish to form buyers' clubs to strengthen their negotiating position with vendors, as described below. These consumers agents use the same profiling techniques described above: given past purchases of a shopper and, optionally, or shoppers with similar profiles, the consumers agent can estimate how much a shopper would be willing to pay for a given offer. If the price of the offer is significantly below the price the shopper is estimated to be willing to pay, then the item is a "bargain" for that shopper. The actual implementation could either take the above form: estimating price as a function of the offer (X) and the shopper (V), or estimating the probability of purchase as a function of the offer and the shopper, and selecting offers with a very high probability of purchase. Such consumer agents could, of course, also be offered by vendors, but with some risk for the shopper that the vendor would not choose to maximize the consumer surplus--i.e., might not find the best bargains for the shopper. In a typical application, the shopper's agent (software with access to both the user's profile and a multiplicity of offers available over the Internet) would examine enormous numbers of offers and select those which the shopper is most likely to purchase. These might include standard repeat purchases (staple items such as food and wine), items where each item is similar but still unique (compact disks and books), and items that are purely novel, but have been purchased by other shoppers with similar tastes

Detail Description Paragraph:
[0292] The same profiling and clustering techniques described above can also be used more generally to match shoppers with vendors or other shoppers who have complementary interests. There are many situations in commerce where it is useful to match up multiple people with similar interests: shoppers can be matched to buy and sell items, to barter and exchange items, to wager with each other about sporting events, to place bids on an item(s) being auctioned to hedge risk, to get lower prices by purchasing in bulk, or to discuss their common interests. A group of shoppers with similar shopper profiles or offer demand summaries can be thought of as a buyers' club or a `mini`-market that is assembled automatically, on an ad hoc basis.

Previous Doc Next Doc Go to Doc#

L4: Entry 1 of 1 File: USPT Oct 15, 2002

DOCUMENT-IDENTIFIER: US 6466918 B1
**** See image for Certificate of Correction ****
TITLE: System and method for exposing popular nodes within a browse tree

Abstract Text (1):
A computer-implemented system and method are provided for identifying popular nodes within a browse tree or other hierarchical browse structure based on historical actions of online users, and for calling such nodes to the attention of users during navigation of the browse tree. The system and method are particularly useful for assisting users in locating popular products and/or product categories within a catalog of an online merchant, but may be used in connection with browse structures used to locate other types of items. Node popularity levels are determined periodically (e.g., once per day) based on recent user activity data that represents users' affinities for such nodes. Such activity data may include, for example, the number of times each item was purchased, and/or the number of times each category was selected for display, within a selected period of time. Popular nodes are called to the attention of users by automatically "elevating" the nodes for display within the browse tree. For example, when a user selects a particular non-leaf category (a category that contains subcategories) for viewing, the most popular items corresponding to the selected category may be displayed (together with the immediate subcategories), allowing the user to view or directly access these items without having to navigate to lower levels of the browse tree (and particularly those associated with leaf categories). Subcategories may be elevated for display in a similar manner. The node elevation process may also be used to elevate items and/or categories that are predicted to be of interest to a user, regardless of popularity. In a preferred embodiment, both popular items are leaf categories are elevated on a user-specific basis using a combination of user-specific and non-user-specific activity data.

Brief Summary Text (2):
The present invention relates to browse trees and other types of hierarchical browse structures used to help users locate online content. More specifically, the invention relates to methods for automatically identifying and calling to the attention of users the nodes (categories and/or items) of a browse tree that are the most popular, or are otherwise predicted to be interesting to users.

Brief Summary Text (5):
Many online merchants and other businesses group their products, services or other items into a set of categories and subcategories of a browse tree. For example, the Yahoo Web site (www.yahoo.com) includes a browse tree which acts as a general Web directory, the Ebay Web site (ebay.com) includes a browse tree for locating auction-related content (auction events, etc.), and the Amazon.com Web site includes a subject-based browse tree for locating book titles.

Brief Summary Text (7):
One problem commonly encountered by online merchants is the inability to effectively present their goods and services to consumers via their browse trees. Due to the large number of items and item categories, many "popular" categories and

items (those that have experienced significant user activity) remain hidden from the user. For example, when a user begins navigation of a typical browse tree for locating books, the user initially sees a list of categories that broadly describe different book subjects. At this point, the user normally would not see more specific categories such as "Olympics," even though "Olympics" may be the most popular category at that time. The "Olympics" category may be nested within the browse tree under Books/Sports & Outdoors/Events/Olympics, requiring the user to navigate downward through multiple levels of the tree to find the category. Similarly, the user would not see the most popular books (e.g., the current bestsellers) because they too would be nested within the browse tree (typically at the lowest level). Further, once the user locates the popular categories and book titles, the user typically has no reason to believe that they are currently the most popular. The ability for users to identify the most popular items and categories helps the users locate items that have gained acceptance within a community or within the population at large.

Brief Summary Text (8):
The present invention addresses these and other problems by providing a computer-implemented system and method for automatically identifying the most "popular" nodes (categories and/or items) within a browse tree or other hierarchical browse structure, and for calling such nodes to the attention of users during navigation of the browse structure. The system and method are particularly useful for assisting users in locating popular products (e.g., books) and/or product categories within a catalog of an online merchant, but may be used in connection with browse structures used to locate other types of items, such as online auctions, chat rooms, and Web sites.

Brief Summary Text (9):
The node popularity levels are preferably determined periodically based on user activity data that reflects users' affinities for particular nodes. The criteria used to measure such popularity levels depend upon the nature and purpose of the browse tree. For example, in the context of a tree used to locate items sold by a merchant, the popularity of each item may be based on one or more of the following, among other, criterion: the number of times the item was purchased, the number of times the item was viewed (within and/or outside the browse tree), the number of times the item was rated or reviewed, and the average rating of the item. The popularity of each category of the same tree may be based on one or more of the following, among other, criterion: the average popularity of the items contained within the category, the number of purchases made within the category relative to the number of items in the category, the number times the category was selected ("clicked through") or searched, and the number of times the category was selected as a destination node of the tree. The specific criteria used within a given system are largely a matter of design choice, and may be varied in order to achieve a particular objective.

Brief Summary Text (10):
The popular nodes are preferably called to the attention of users by automatically "elevating" the nodes along child-parent paths for display within the browse structure. For example, when the user selects a particular non-leaf category (a category that contains subcategories) for viewing, the most popular items corresponding to the selected category may be displayed together with (e.g., on the same Web page as) the immediate subcategories, allowing the user to view or directly access these items without navigating to lower levels of the browse tree. Subcategories may be elevated for display in a similar manner.

Brief Summary Text (15):
In an embodiment for use by an online bookseller, the system and method are used to "feature" the most popular book titles and leaf categories on Web pages corresponding to higher-level categories. The most popular books and categories are preferably determined periodically based on purchase counts, category click-through

rates, and/or other types of user activity data. The nodes to be featured are preferably selected recursively, on a node-by-node basis, by selecting the most popular nodes from the immediate children of the current node. Books and low-level categories that are currently very popular thus tend to be featured at many different levels of the tree, increasing the probability of exposure in proportion to level of popularity. Preferably, the nodes are selected for elevation based on a combination of user-specific and collective user activity data, so that the featured books and categories reflect both the interests of the particular user and the interests of others.

Brief Summary Text (17):
The invention may also be used to highlight personal recommendations of items that exist within the browse tree. For example, an item may be selected from the tree for personal recommendation using a collaborative filtering, content-based filtering, or other recommendations algorithm, and automatically featured at some or all of the categories in which the item falls. Alternatively, the criteria and methods used to generate personal recommendations may simply be incorporated into the algorithm for generating item popularity scores.

Drawing Description Text (3):
FIG. 1A illustrates an example Web page which includes a set of featured book categories and a set of featured book titles that have been elevated for display.

Drawing Description Text (5):
FIG. 2 illustrates a set of Web site components that may be used to identify and elevate book categories and titles within a browse tree according to the invention.

Drawing Description Text (6):
FIG. 3 illustrates a method for generating a table of the top book titles (items) within each leaf category.

Drawing Description Text (7):
FIG. 4 illustrates a method for generating scores that represent user-specific and collective popularity levels of specific leaf categories.

Drawing Description Text (14):
FIG. 11 illustrates a method for recursively selecting, for a particular user, the top titles corresponding to each non-leaf category of the browse tree.

Detailed Description Text (2):
A system which represents a preferred embodiment and example application of the invention will now be described with reference to the drawings. Variations to this system which represent other preferred embodiments will also be described. In the disclosed system, the invention is used to automatically identify book titles and low-level book categories to be featured at higher levels of a browse tree of an online bookseller. It will be recognized, however, that the invention is also applicable to browse trees used to help users locate other types of categories and items, including but not limited to authors, news articles, online auction items, other types of products, sound clips, downloadable software, chat rooms, classified ads, restaurants, stores, multimedia channels, and other Web sites. Although the invention is used in the disclosed system to feature both categories and items (book titles), it should be understood that, in other embodiments, only one of these two types of nodes, or a different type of node, could be featured.

Detailed Description Text (5):
The various book titles that are available for purchase through the bookseller's Web site are arranged within various categories and subcategories of a browse tree. Users of the Web site can navigate the browse tree to locate books titles (the "items" of the browse tree) based on various pre-defined subjects and other

classifications. Users can also locate books of interest using the site's search engine, recommendation services, and other types of navigational aids. Users can also submit reviews and ratings of books they have read.

Detailed Description Text (6):
The browse tree is preferably in the form of a directed acyclic graph (a tree that allows a child node to have multiple parents), although a pure tree or other type of browse structure could be used. The lowest-level nodes (or "leaf-nodes") of the browse tree represent individual book titles, and all other nodes represent categories (including sub-categories) of books. The lowest-level categories (those with no subcategories) are referred to herein as "leaf categories." Each node is preferably displayed to the user as a hyperlink (see FIG. 1A), although other types of user interfaces could be used.

Detailed Description Text (8):
The categories may include pre-existing categories that are used within the industry and/or categories that are created for purposes of implementing the invention.

Detailed Description Text (9):
The categories may alternatively be selected or modified dynamically (automatically and/or by system administrators) based on user actions and other criteria. Table 1 illustrates an example set of top-level book categories that may be used in one embodiment. As illustrated by Table 1, the book categories are primarily in the form of subject and genre classifications.

Detailed Description Text (10):
Further, the categories are preferably selected so as to encompass a reasonably wide range of related user interests. Each category may lead the user to another set of subcategories. For example, when a user selects the "Sports & Outdoors" top-level book category, the user may be led to another set of book categories similar to those shown in Table 2. This second level of categories can also have a set of subcategories, and so forth, creating a tree-like structure. In the preferred embodiment, the categories are not mutually exclusive (i.e., a book can fall within multiple categories, and/or a subcategory can fall within multiple categories), although mutually exclusive categories and items can alternatively be used.

Detailed Description Text (11):
Preferably, each category and item has a unique name that can be displayed to the user. For example, while many book subcategories may appear on the Web page as a "General" link, the actual link refers to the complete book category name such as "Sports & Outdoors--Skiing--General" or "Mystery--General" which could also be displayed.

Detailed Description Text (12):
In accordance with a preferred embodiment of the invention, the Web site system includes software and database components that are used to collect information about the browsing and/or purchasing activities of users, and to use this information to automatically evaluate the popularity levels of specific item nodes and category nodes of the tree. Nodes that are determined to be the "most popular" are automatically elevated for display or "featured" (as described below) at higher levels of the tree. In the preferred embodiment, the only types of categories that are featured are the leaf categories, although higher level categories could be featured in other embodiments. Node popularity levels are preferably determined based on user activity data falling within a sliding window (e.g., data collected over the last two weeks), so that the featured nodes strongly reflect the current trends and fads.

Detailed Description Text (13):
The data collected for the category nodes may include, for example, the number of

purchases made within each category, the number of searches performed within each category, click-through counts (the number of times each node was selected by a user), and/or other types of activity data. Where click-through counts are used, click through events that do not result in a purchase, and/or do not represent the user's final destination, may be disregarded or given a lesser weight.

Detailed Description Text (14):
In one embodiment, such data is collected only for the leaf categories, since higher-level categories are not elevated for display. In another embodiment, the data is also collected for the non-leaf categories and is used to "weight" popular items lists (see FIG. 11) during selection of featured books. The data collected for the item nodes preferably includes purchase data (the number of times each item was purchased), and may additionally or alternatively include other types of data such as the number of times each item was viewed, rated, reviewed, or placed into a online shopping cart.

Detailed Description Text (15):
The popularity levels of the nodes can be determined by evaluating the collected data on a collective basis (without regard to user identity), on an individual basis, or both. Where only collective evaluation is performed, the items and leaf categories that are featured at any given node of the tree are the same for all users. Where the data is collected and evaluated on an individual basis, the items and leaf categories that are featured at each node are specific to the historical actions performed by the particular user. For example, the popularity levels may reflect the user's affinities for particular items as predicted by a collaborative filtering, content-based filtering, or other algorithm for generating personal recommendations. An example of a recommendation algorithm that can be used for this purpose is described in U.S. patent application Ser. No. 09/157,198, filed Sep. 18, 1998, the disclosure of which is incorporated herein by reference. In the embodiment set forth below, a combination of collective and individual evaluation is used, so that the featured nodes are dependent upon both the actions of the particular user and the actions of the community as a whole.

Detailed Description Text (17):
FIG. 1A illustrates an example Web page that includes an example set of featured book categories 110 and featured book titles 120. As depicted by the figure, the "featured" book categories 110 and "featured" book titles 120 are derived from the "Sports & Outdoors" branch of the browse tree which is the branch currently selected for viewing. For example, the category "Olympics" is featured even though it is actually found under the following path: Books.backslash.Sports & Outdoors.backslash.Events.backslash.Olympics, and the book "Wayne Gretsky: A Hockey Hero" is featured even though it would be found under the following path: Books.backslash.Sports & Outdoors.backslash.Hockey. The featured books and categories are displayed as respective hyperlinks that provide a direct path to the corresponding books and categories. This gives the user quicker access to the most popular leaf categories and books. For example, selection of a link for a featured book causes the book's detail page to be displayed, and selection of a link for a featured leaf category causes the list of books falling under that category to be displayed.

Detailed Description Text (18):
The Web page also provides links 130 to the immediate subcategories of the selected book category in alphabetical order. Although the featured items and categories are featured explicitly in FIG. 1A, they could alternatively be featured implicitly as regular entries on the page. For example, the featured leaf categories 110 and could simply be displayed as part of the list 130 of subcategories.

Detailed Description Text (19):
As the user moves further into the browse tree, the "featured" book categories and book titles adjust such that the most popular leaf categories and book titles

falling within the selected category are displayed. Preferably, the featured books are displayed as such only at levels of the tree at which the book titles are not visible, and featured categories are displayed as such only at levels at which leaf categories are not visible. Thus, the effect is to expose to the user, or to "elevate" within the tree, popular book titles and categories that would not otherwise be visible at the current level. Elevation preferably occurs only along child-parent paths, so that a node will only be featured in association with its parent nodes. In the preferred embodiment, the elevated nodes can also be accessed by navigating downward to the "fixed" positions of such nodes. Thus, the process of elevating popular nodes preferably involves copying, as opposed to moving, the nodes to higher levels of the tree. In other embodiments, the nodes may actually be moved within the browse tree.

Detailed Description Text (20):
When the user selects a leaf category to view a corresponding list of book titles, the most popular book titles within that category may optionally be highlighted (not illustrated), such as by displaying them at the top of the list or in a particular color. Similarly, when the user selects a category that contains only leaf categories, the most popular leaf categories in the list may optionally be highlighted (not shown) in the same or a similar manner.

Detailed Description Text (21):
In one embodiment, the leaf categories and book titles to be featured (elevated) are automatically selected based upon a popularity score which reflects activity from a collection of users as well as activity from the specific user viewing the page. As indicated above, the nodes may alternatively be elevated based solely on one of these two classes of user activity. In addition, the nodes could be elevated based in-whole or in-part on the actions of the members of one or more communities to which the user belongs. The score preferably gives more weight to activities that are deemed the most indicative of users' affinities for specific categories and items. For example, an actual purchase of an item is preferably given more weight than merely placing the item in the shopping cart. In addition, activity from the current user is preferably given more weight than activity of other individual users.

Detailed Description Text (22):
In addition to node popularity levels, other types of criteria may be used to select the nodes to be elevated. For example, a bias can be added to node selection process to cause newly added items and/or leaf categories to be elevated more frequently than other types of nodes.

Detailed Description Text (23):
As described below, the task of processing historical data to evaluate book and category popularaties is preferably performed offline (i.e., not in response to page requests), and the results stored in one or more tables or other data structures. This allows the featured book titles and categories to be selected for each user in a timely manner. In other embodiments, however, some or all of such processing can be performed in real-time in response to page requests.

Detailed Description Text (25):
FIG. 1B illustrates a simple browse tree, and will be used to describe a preferred process for elevating items for display. The same method may be used to elevate categories. The tree consists of seven category nodes, C1-C7, and fifteen item nodes, I1-I15. The numbers listed below the item nodes ("items") are their respective popularity scores, on a scale of 1-10. As indicated above, these scores may be based on activity data collected for a particular user, a set of communities of which the user is a member, the general user population, or a combination thereof.

Detailed Description Text (26):

Assuming that the top two items (items with the highest scores) are selected for elevation at each category node, the items are elevated for display as shown to the right of each category node. For example, items 5 and 6 are elevated for display at category 5 since they have the highest scores of all items falling within category 5; and items 9 and 10 are elevated for display at category 3 since they have the highest scores of all items falling within category 3. In this example, items 1 and 5 would be featured both at the root of the tree (e.g., a Web page which lists the top level categories C2 and C3) and at category C2 (e.g., a Web page which lists C4 and C5), and items 9 and 10 would be featured at category C3. When the user navigates down to one of the leaf categories C4-C7 to view a list of items, the elevated items within that category might be highlighted within the list.

Detailed Description Text (27):
As indicated above, a recursive process is preferably used to elevate the nodes within the tree. Table 3 is a pseudocode representation of one such algorithm that may be used to elevate category nodes (referred to as "browse nodes" in Tables 3 and 4). Table 4 is a pseudocode representation of a more generic recursive algorithm that may be used to elevate category nodes or item nodes. The term "item" is used generically in Tables 3 and 4 to refer to both types of nodes.

Detailed Description Text (30):
The Web site 210 includes various server components 220, including a Web server (not shown), that are used to process requests from user computers 230 via the internet 240. The server components 220 access a database of HTML documents 250, a Bibliographic Database 260, a User Database 270, and a Browse Tree Component 280. The Bibliographic Database 260 includes records for the various book titles and other products that are available for purchase from the Web site. The Bibliographic Database 260 also includes information regarding the set of existing categories, how the categories are related to each other, and the categories in which each book title falls..

Detailed Description Text (31):
The User Database 270 includes information about the users of the site and keeps track of their activity. As depicted by FIG. 2, the information stored for each user may include the user's purchase history 272 (if any) and the user's Web activity 274 (if any), and a list of the communities of which the user is a member. The purchase histories 272 keep track of the products that have been purchased by the user and may, for example, be in the form of lists of product identification numbers (such as ISBNs of books) and corresponding dates of purchase. The Web activity 274 keeps track, on a user-specific basis, of certain types of browsing events, such as downloads of book detail pages, book rating events, selections of items for placement in the shopping cart, searches within specific categories, etc. The Web activity data may alternatively be tracked only on a community-specific basis, without regard to user identity. The executable components used to process orders, update the purchase histories and Web activity data, implement shopping carts and the search engine, and perform other sales-related tasks are omitted from FIG. 2 to simplify the drawing.

Detailed Description Text (33):
The Browse Tree Component 280 includes a Table Generation Process 282, a Featured Nodes Selection Process 284, and a Request for Browse Tree Page Process 286. (The term "process," as used herein, refers generally to a computer program stored in a computer memory, and is also used to refer to the method implemented by the computer program.) The Table Generation Process 282 uses the purchase history and Web activity data to generate a Category Popularity Table 290 and an optional Popular Items Table 292. Other types of data structures may be used in place of the tables 290, 292.

Detailed Description Text (34):
As depicted in FIG. 2, the Category Popularity Table 290 preferably contains a

popularity score for each (user, category) pair. This score represents the user's predicted interest in the category based on the user's previous activities. Such scores (referred to as "individual user history scores" or "individual scores") may be generated, for example, for every known user of the Web site, or for a selected subset of users that visit the site on a frequent basis. An algorithm for generating personal recommendations may be used to generate the individual scores. Scores for the non-leaf categories may optionally be omitted. In addition, in embodiments in which featured categories are not selected on a user-specific basis, the individual user history scores may be omitted.

Detailed Description Text (36):
The table 290 also includes popularity scores for the general population, referred to herein as "collective user history scores" or simply "collective scores." The Table Generation Process 282 updates the table 290 periodically, such as once per day, so that the scores strongly reflect the current interests of users. In one preferred embodiment, which is depicted in FIGS. 5-8, the scores are based on several different types of user activities. In other embodiments, the individual and collective scores are based solely on a particular type of activity, such as purchases or click-through counts. As described below, the individual and collective scores are preferably used in combination to select leaf categories for elevation on a user-specific basis. In one embodiment (not illustrated), the table 290 also stores a popularity score for each (community, category) pair, and these community-specific scores are incorporated into the total scores based on community memberships of users.

Detailed Description Text (37):
Because the number of items contained within the catalog is large (several million items), a Popular Items Table 292 is used in the illustrated embodiment to store item popularity data. As depicted in FIG. 2, this table 292 contains a list of the most popular items (e.g., the ten most popular items) within each leaf category (CAT1, CAT2 . . .). Popular items lists for non-leaf categories may optionally be stored in the table 292 as well. The Table Generation Process 282 preferably generates these popular items lists periodically from purchase history data, and possibly other types of activity data. Each item within each popular items list is preferably stored together with a weight value (not shown) that indicates the popularity of the item. As described below, the popular items lists are preferably used in combination with the individual and collective scores to select items for elevation on a user-specific basis. One benefit to this approach is that it provides customized (user-specific) elevation of items without the need to generate individual scores for the items.

Detailed Description Text (38):
In a second embodiment (not illustrated), the Popular Items Table 292 is omitted, and table 290 is supplemented with the individual and collective scores for some or all of the items in the tree. In this second embodiment, a common node elevation algorithm of the type shown in Table 4 is used to elevate both types of nodes (categories and items).

Detailed Description Text (39):
The Featured Nodes Selection Process 284 uses the information stored in the tables 290 and 292 to select the leaf categories and book titles to be displayed (featured) at higher-level nodes of the browse tree. As indicated above, the featured categories and book titles are preferably selected on a user-specific basis. In one embodiment, the leaf categories and book titles to be displayed to a given user at each higher-level node are determined and are stored in a temporary table when the user initiates a browsing session or begins using the browse tree, and this temporary table is accessed when the user requests an appropriate browse tree page. Thus, the processing and storage burden associated with elevating nodes is avoided for those users who do not access the site or the browse tree during the relevant time period to which the scores correspond. The identity of the user may

be determined using cookies, a log-in procedure, or another appropriate identification method.

Detailed Description Text (42):
In block 330, the process uses the purchase counts generated in block 320 to identify the best-selling Y items (e.g., 10 items) in each leaf category. Each such list of best-selling items, together with the corresponding purchase count values, is then recorded in the table 292 as a popular items list. The method of FIG. 3 can optionally be extended to generate popular items lists for non-leaf categories.

Detailed Description Text (44):
IV. Generation of Category Popularity Table

Detailed Description Text (45):
FIG. 4 illustrates an algorithm that may be used by the Table Generation Process 282 to generate the Category Popularity Table 290. This algorithm is preferably applied to the collected purchase history and Web activity data periodically (e.g., once per day) to generate new table data. As will be apparent from the description, the same or a similar algorithm could be used to score items.

Detailed Description Text (46):
In block 410, the process 282 retrieves the purchase history and Web activity data from the User Database 270. In block 420, the process uses this data to generate individual user history scores for each (user, leaf category) pair, and stores the resulting scores in the Category Popularity Table 290. The details of block 420 are set forth in FIGS. 5 and 6 and are discussed below. In embodiments in which non-leaf categories are elevated, scores may also be generated for the non-leaf categories.

Detailed Description Text (48):
In block 430, the process 282 evaluates the purchase history and Web activity data on a collective basis to generate the collective scores for each leaf category, and stores the resulting scores in the table 290. The details of block 430 are illustrated in FIGS. 7 and 8 and are discussed below. In an alternative embodiment, the collective scores are generated by summing the individual scores within each leaf category. In embodiments in which non-leaf categories are elevated, collective scores may also be generated for the non-leaf categories.

Detailed Description Text (50):
As illustrated in FIG. 5, to generate the individual user history scores for each (user, category) pair, the purchase history and Web activity data (collectively "user history") are processed on a user-by-user basis (blocks 505 and 560). First, an individual user history is retrieved (block 510). Next, the individual user history information is preferably restricted to user activity performed within a sliding window (block 515), such as the most recent three months. This window size could be selected dynamically based on the quantity of recent purchase history data available for the user. The book category count for each type of user activity is then initialized to zero (block 520). For each book purchased by the user, the "Purchase" count for each book category in which the book falls is incremented (block 525).

Detailed Description Text (51):
For each book category the user "clicked-through" during browsing of the tree, the "Click-Through" count is incremented (block 530). Click-through events that did not result in purchases, and/or did not represent the user's destination, may be ignored. Well-known log tracing techniques may be used to determine the user's actions following a click-through event.

Detailed Description Text (52):
For each book category in which the user performed a search, the "Search" count is

incremented (block 535). For each book that the user rated, the "Rating" count is incremented (block 540) for all book categories in which the book falls. For each book placed in the shopping cart, the "Shopping Cart" count is incremented for each book category in which the book falls (block 545). In other embodiments, other user activity may also be counted. In addition, any one of the foregoing types of activity, or a different type of activity, could be used as the exclusive indicator of item popularity. As with the FIG. 3 algorithm, the count values may be generated only for the period of time since the last execution of the algorithm (e.g., the last day), and the results combined with prior results data.

Detailed Description Text (53):
Once the process 282 has gone through all relevant user activity, the process calculates a final score based upon predetermined weights for each book category count (block 550).

Detailed Description Text (54):
The predetermined weights reflect preferential user activity. For example, actual purchases are preferably given more weight than merely placing an item in a shopping cart. The weights may be adjusted by system administrators to "tune" the system. Table 5 illustrates sample weights for some types of individual user activity. Other weights or an equally-weighted approach could be used. Table 6 illustrates how the score is calculated for each category using the predetermined weights of Table 5.

Detailed Description Text (55):
Next, the weighted scores (individual user history scores) for the user are stored in the Category Popularity Table 290 (block 555). The scores could alternatively be stored in the User Database 270 (as part of the user's profile), as a cookie stored by the user's computer, or elsewhere. The process then moves on to the next individual user history and repeats until it reaches the last individual user history (blocks 505, 560).

Detailed Description Text (57):
As illustrated in FIG. 7, to generate the collective user history scores, first, the book category count for each type of user activity is initialized to zero (block 705). Then, an individual user history is retrieved (block 715). Next, the individual user history information is preferably restricted to user activity performed within a sliding window (block 720), preferably the most recent two weeks. The use of a shorter window than the window used for individual scores is justified by the greater quantity of data used to generate the collective scores. In other embodiments, other restrictions can be added. For example, the process could restrict the set of individual user histories to those whose ages are over fifty-five or to user activity performed after midnight. Next, for each book purchased by the user, the "Purchase" count is incremented for each book category in which the book falls (block 725). For each book category the user "clicked-through," the "Click-Through" count is incremented (block 730). For each book category in which the user has performed a search, the "Search" count is incremented (block 735). For each book that the user rated, the "Rating" count is incremented for each book category in which the book falls (block 740). For each book placed in the shopping cart, the "Shopping Cart" count is incremented (block 745) for each book category in which the book falls. In other embodiments, the process could also account for other user activity, or could use only a subset of the types of activity listed in FIG. 7.

Detailed Description Text (59):
After all user histories have been traversed, a final score is calculated (block 755) based upon predetermined weights for each book category.

Detailed Description Text (60):
As with the individual scores, the predetermined weights used for collective scores

reflect preferential user activity and can be adjusted by system administrators to tune the system. Table 7 illustrates sample weights for some types of collective user activity. The collective user activity is preferably weighted less than individual user activity. It is recognized, however, that other weights or an equally-weighted approach could be used. In addition, the scores could alternatively be based solely on a particular type of activity such as click-through events. Table 8 illustrates how the score is calculated for each category using the predetermined weights of Table 7.

Detailed Description Text (61):
Next, the weighted scores (collective user history scores) are stored in the Category Popularity Table 290. FIG. 8 shows an example set of scores 810 after the process has finished. As illustrated, the weighted scores in FIG. 8 correspond to those calculated in Table 8. Preferably, only the weighted scores are stored in the table 290, and the other scores are stored temporarily during generation of the weighted scores.

Detailed Description Text (63):
V. Elevation of Leaf Categories and Book Titles

Detailed Description Text (64):
The Feature Nodes Selection Process 284 may be initiated when a user performs a particular type of action, such as initiating a browsing session or requesting a page of the browse tree. For example, the nodes to be featured may be determined for the entire tree (using the previously-generated scores) when the user initially accesses the tree, and the results cached in a table or other data structure during the browsing session. As the user navigates the browse tree, this table may be accessed to look up the featured categories and books. The categories and books to be featured could alternatively be determined off-line whenever new scores become available.

Detailed Description Text (65):
As depicted by FIG. 9, the first step of the selection process involves combining the user's individual user history scores (if any) with corresponding collective user history scores to generate total scores. If no individual scores exist for the user, a set of default individual scores may be used, or the collective scores may be used as the total scores. As described below, the total scores are subsequently used to identify leaf categories and book titles to be elevated. In blocks 910 and 920, the user's individual scores and the collective scores are retrieved from the Category Popularity table 290. Then for each entry, the individual user history score is combined with the collective user history score (block 930). In other embodiments, the process may give more weight to the individual user history scores. The results are stored in a temporary table or other data structure (block 940). FIG. 10 shows how the individual and collective scores are combined for an example set of values.

Detailed Description Text (66):
In a first embodiment, the method shown in FIG. 9 is applied only to the leaf category nodes, and not to the item nodes. One of the recursive algorithms shown in Tables 3 and 4 is then used to elevate the category nodes, and the process shown in FIG. 11 (described below) is used to elevate the items nodes. One benefit of this first embodiment is that it does not require individual or collective scores to be generated for the items in the tree. In a second embodiment, in which collective and individual scores are also generated for the items, the FIG. 9 method is applied to both types of nodes (items and categories), and the recursive algorithm shown in Table 4 is used to select both types of nodes for elevation.

Detailed Description Text (67):
FIG. 11 illustrates an algorithm that may be used to identify the most popular items (book titles) corresponding to each category node of the tree without the

need to generate individual or collective scores for the items. Because the most popular items corresponding to the leaf categories are already known (i.e., are stored in the Popular Items Table 292), this algorithm is preferably applied only to the non-leaf categories. The algorithm operates recursively, starting at the lowest applicable level, and proceeding successively to higher levels until the last node is reached. To customize the featured items to the particular user, the total scores generated by the FIG. 9 process are used to weight the popular items lists. As indicated above, the featured items could alternatively be selected without regard to user identity.

Detailed Description Text (68):
In block 1110, the process obtains the popular items lists for each immediate child node of the current node. If the immediate child is a leaf category node, the popular items list is read directly from the Popular Items Table 292; otherwise, the popular items list is obtained from a temporary table generated from previous iterations of the FIG. 11 process. As depicted in block 1110, each list is preferably weighted based on the total score for the respective child to customize the selection process for the particular user. This may be accomplished, for example, by multiplying the total score by the weight value of each item in the list. In block 1120, the weighted lists are combined while summing the weights of like items. The Y items with the highest weights are then stored in a temporary table as the popular items list for the current node (block 1130). If the user selects this node for viewing during the browsing session, some or all of these Y items may be displayed as featured book titles. The process then proceeds to the next category (not shown), or else terminates if the root node has been reached.

Detailed Description Paragraph Table (4):
TABLE 4 // A more complex (but more generic) alogithm to surface browse nodes // or individual category elements. surface_list_of_relevant_items (node_id, list_of_items, type_of_item_to_surface) { if (is_a_leaf_node (node_id) { if (type_of_item_to_surface == browse_node) populate_list_with_browse_node_and_score (list_of_items, node_id, get_score (node_id)) if (type_of_item_to_surface == item_elements) populate_list_with_top_scoring_elements (list_of_items, top_items_with_score node_id)) }else { // If not a leaf node for each child_id of node_id { surface_list_of_relevant_items (child_id, list_of_items, type_of_item_to_surface) } }

Detailed Description Paragraph Table (6):
TABLE 6 Click- Shopping Book Purchase Through Search Rating Cart Weighted Category (210) (201) (203) (208) (207) Scores Air Sports & (0 * 210) + (0 * 201) + (0 * 203) + (0 * 208) + (0 * 207) = 0 Recreation Audiobooks (3 * 210) + (24 * 201) + (35 * 203) + (3 * 208) + (7 * 207) = 14632 Automotive (0 * 210) + (19 * 201) + (21 * 203) + (0 * 208) + (0 * 207) = 8082 . . . Reference (0 * 210) + (0 * 201) + (0 * 203) + (0 * 208) + (0 * 207) = 0 Scuba (0 * 210) + (0 * 201) + (0 * 203) + (0 * 208) + (0 * 207) = 0 Swimming (8 * 210) + (73 * 201) + (57 * 203) + (12 * 208) + (6 * 207) = 31662 Yoga (0 * 210) + (0 * 201) + (0 * 203) + (0 * 208) + (0 * 207) = 0

Detailed Description Paragraph Table (8):
TABLE 8 Click- Shopping Book Purchase Through Search Rating Cart Weighted Category (10) (1) (3) (8) (7) Scores Air Sports & (564 * 10) + (616 * 1) + (1055 * 3) + (61 * 8) + (57 * 7) = 16009 Recreation Audiobooks (2016 * 10) + (8465 * 1) + (2461 * 3) + (248 * 8) + (189 * 7) = 39315 Automotive (5354 * 10) + (7715 * 1) + (3403 * 3) + (1127 * 8) + (2092 * 7) = 95124 . . . Reference (715 * 10) + (946 * 1) + (1035 * 3) + (183 * 8) + (247 * 7) = 14394 Scuba (226 * 10) + (546 * 1) + (887 * 3) + (311 * 8) + (302 * 7) = 10069 Swimming (3452 * 10) + (4652 * 1) + (4512 * 3) + (415 * 8) + (521 * 7) = 59675 Yoga (1530 * 10) + (765 * 1) + (996 * 3) + (534 * 8) + (454 * 7) = 26503

CLAIMS:

1. A computer-implemented method for facilitating identification of popular items within an electronic catalog of an online store, comprising: providing a browse tree which contains a plurality of nodes arranged in a plurality of levels, wherein leaf nodes represent items of the catalog and non-leaf nodes represent categories of items, said browse tree being browsable by users to locate items according to item categories and subcategories; processing at least the purchase history data to generate node popularity data; and based at least in-part on the node popularity data, automatically elevating selected items for display to expose popular items to users during viewing of corresponding non-leaf categories of the browse tree; whereby popular items are brought to the attention of users during browsing of non-leaf categories to which such popular items correspond.

6. The method of claim 1, wherein elevating comprises featuring popular items on a Web page that corresponds to a non-leaf category.

14. The method of claim 1, further comprising elevating selected leaf categories for display based at least in-part on popularity levels of such leaf categories.

Previous Doc Next Doc Go to Doc#